Design, Development, and Reuse of Software Agents in the Knowledge Management for Distributed Tracking (KMDT) Program

Dave Swanson, Tedd L. Wright, Marion G. Ceruti, Ph.D.*
Dwight R. Wilcox, and Scott C. McGirr, Ph.D.
Space and Naval Warfare Systems Center, San Diego (SSC-SD), Code 246206*
53560 Hull Street, San Diego, CA 92152-5001
dave.swanson@navy.mil, tedd.wright@navy.mil, marion.ceruti@navy.mil,*
dwight.wilcox@navy.mil, scott.mcgirr@navy.mil

Abstract. Knowledge Management for Distributed Tracking is an ongoing research and development program to improve naval command, control, and decision support. The program's approach includes modeling and simulation, intelligent software agents, integrated-sensor ontology, and reusing methods from past projects. The software simulates a hypothetical scenario designed to demonstrate how knowledge-management technologies can improve situation awareness and reduce information overload. In the simulation, when additional information is desired about an unknown contact, intelligent-software agents are deployed on a secure, web-like network to find sensor data collected at other friendly platforms. The information is fused to reduce the uncertainty in the detection, localization, tracking, classification and identification of the unknown contact. The paper describes how the agents interact with the integrated sensor ontology, which facilitates distributed, heterogeneous sensor-data fusion.

Key words: Decision-support system, distributed agent-based system, knowledge acquisition and management, modeling and simulation, integrated sensor ontology, sensor-data fusion, tracking

1 Introduction

The Knowledge Management for Distributed Tracking (KMDT) Program supports the concept of network-centric warfare as envisioned in FORCEnet [1] [2], which is the U.S. Navy's operational construct and architectural framework for naval warfare in the information age. To accomplish this, a distributed tracking problem was simulated for subsequent demonstration and analysis that primarily involves the localization of targets via intersection of Lines-of-Bearing (LOBs) from distributed passive sensors [3]. Although highly simplified, this problem may reflect current political and operational strategies that restrict usage of active systems for environmental and security reasons. The simplification is intended to focus the demonstration on the advantages of a network-centric concept of operations over those involving legacy stand-alone architectures and systems.

KMDT is focused on technologies that are essential for the design of next-generation tracking systems that use knowledge-management techniques and network-based command and control to reduce uncertainty in command decisions. These technologies include distributed sensing, modeling and simulation, [3], intelligent agents [4], [5], [6], and ontology [7], [8], [9]. Distributed tracking can revolutionize traditional level-one data fusion (e.g. detection, localization, tracking, classification, and identification [2], [8], [10]). Today, the Navy does not use the majority of sensor data due to lack of correlation opportunities. Through knowledge management as modeled in the KMDT simulation, some of these unused sensor data that are now unavailable can be fused in a timely manner to localize, track and classify

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE
28 JUN 2007

2. REPORT TYPE
N/A

3. DATES COVERED
-

4. TITLE AND SUBTITLE
Design, Development, and Reuse of Software Agents in the Knowledge Management for Distributed-Tracking (KMDT) Program Published in Proceedings of the 2nd International Workshop on Ontology, Conceptualization, and Epistemology for Software & Systems Engineering (ONTOSE 2007) 27-28 June 2007, Milano Italy.

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
David Swanson, Tedd L. Wright, Marion G. Ceruti, Dwight R. Wilcox, and Scott C. McGirr

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
SSC Pacific San Diego, CA 92152-5001

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release, distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT
Knowledge Management for Distributed Tracking is an ongoing research and development program to improve naval command, control, and decision support. The programs approach includes modeling and simulation, intelligent software agents, integrated-sensor ontology, and reusing methods from past projects. The software simulates a hypothetical scenario designed to demonstrate how knowledge management technologies can improve situation awareness and reduce information overload. In the simulation, when additional information is desired about an unknown contact, intelligent-software agents are deployed on a secure, web-like network to find sensor data collected at other friendly platforms. The information is fused to reduce the uncertainty in the detection, localization, tracking, classification and identification of the unknown contact. The paper describes how the agents interact with the integrated sensor ontology, which facilitates distributed, heterogeneous sensor-data fusion.

15. SUBJECT TERMS
Decision-support system, distributed agent-based system, knowledge acquisition and management, modeling and simulation, integrated sensor ontology, sensor-data fusion, tracking

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			
unclassified	**unclassified**	**unclassified**	**UU**	**12**	

Targets Of Interest (TOI). This will result in a future reduction in uncertainty in command centers.

The distributed KMDT architecture enables any operator connected to the network to deploy intelligent agents to retrieve sensor data posted to common repositories at ship and shore-based sensor-data collection stations. The network does not contain a single fusion center where all fusion processing takes place, but rather, each ship or shore-based sensor station can perform distributed sensor-data fusion as needed to meet their local requirements. The agents retrieve processed sensor data at the message level and are not connected directly to the remote sensors themselves.

KMDT's Modeling and Simulation (M&S) approach uses a hypothetical scenario to develop and evaluate knowledge-management techniques [3]. M&S helps to conceptualize the salient features of the battle space and to focus on the main characteristics of interest, such as how ships move in the water and how sensors detect them, in a controlled environment that is not available in the physical world. It also allows the testing of many hypothetical scenarios at low cost. When the concepts are demonstrated using simulated experiments, these results serve collectively as a point of departure for more realistic field tests.

Unlike older legacy systems that rely primarily on similar sensor types to detect, localize, and track TOI, new approaches also can use dissimilar sensor types for level-one data-fusion tasks. Our previous research was oriented toward the simulation details [3] and the integrated sensor ontology [7], [8], [9] as separate components of KMDT. However, the present work is more focused on software reuse in the agent design, the agent-ontology interaction, and the database that instantiates some of the concepts in the integrated sensor ontology. Software reuse enables a low-cost applied research and development in the KMDT program. For example, the initial KMDT integrated sensor ontology was based on ontology components developed from previous projects, as explained in ontology [7], [8], [9].

This paper is organized as follows. Section 2 describes the architecture. Section 3 presents the simulation design. Section 4 explains static and dynamic database design vis-à-vis the integrated sensor ontology. Section 5 describes the intelligent software agent design and development. Section 6 describes software reuse in agent design. Section 7 discusses agent-ontology interactions. Section 8 describes results. Section 9 suggests future directions.

2 Architecture

The KMDT functional configuration, described in detail in [3], employs two computers connected via a secure Local Area Network (LAN). The first computer, called the "simulation computer," implements scenarios involving multiple targets and sensors. It generates LOB alerts on detected targets as well as false alarms, displays them, and posts them to the respective dynamic-detection database that it maintains independently for each sensor. Maintaining independent access is necessary to represent sensors distributed among different platforms.

Requests for information originate in the second computer called the "command-center computer." It executes several kinds of intelligent agents that gather and subsequently fuse appropriate information obtained from the network. The network environment is simulated by access, via the LAN, to the dynamic detection database maintained on the simulation computer. Fig. 1 illustrates the functional architecture arranged in the sequence of events not only for the hypothetical operational use of agents but also for conducting simulated agent experiments to verify and validate the accuracy and completeness of the agent behavior as measured by statistics summarizing the results.

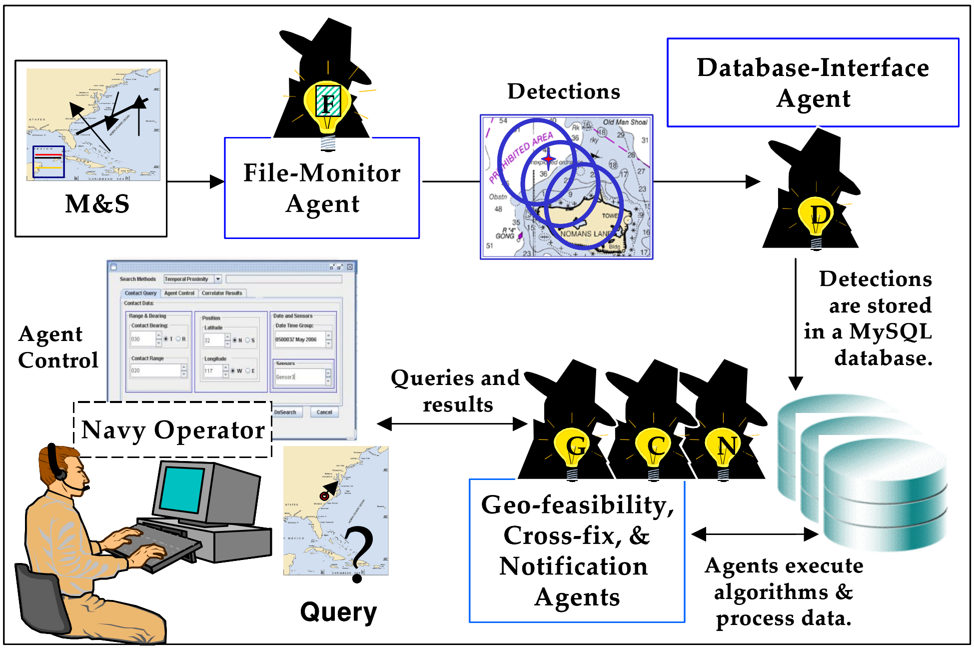

Fig. 1. KMDT's intelligent-agent functional architecture showing the types of agents and their tasks.

3 Modeling and Simulation Design

The simulation, an earlier version of which was described in [3], is intended to represent realistic trial scenarios without conducting costly field experiments. Additionally, it provides an analysis tool to quantify results. The primary components of the simulation consist of a scenario and a contact generator. The simulation scenario consists of predefined platforms in the theater of operations, with specified capabilities, sensors, and initial locations. For the demonstration, the capabilities and performance (C&P) parameters of the simulated sensors will be defined in a static C&P database. The architecture for the Navy's implementation of the technology that the simulation is designed to demonstrate calls for the sensors' C&P parameters to be communicated either on the web page that represents each sensor, or as a distributed database.

Additionally, the scenario contains multiple targets with specified capabilities and initial locations. As the simulation progresses, the positions of the targets and mobile sensor platforms are updated via appropriate motion models at each time step of the simulation. To support the detection decision within the contact generator, the range and LOB from each platform to each target are computed. A sensor-performance model determines the probability of detection computed as a function of this range. Detections are declared when a random number selected from a uniform distribution between 0 and 1 is less than the probability of detection. False alarms are generated similarly from acceptable probabilities of false alarm for each sensor. When the simulation declares either a detection or false

alarm, information generated by the simulation is posted to dynamic-detection databases described in section 4 below.

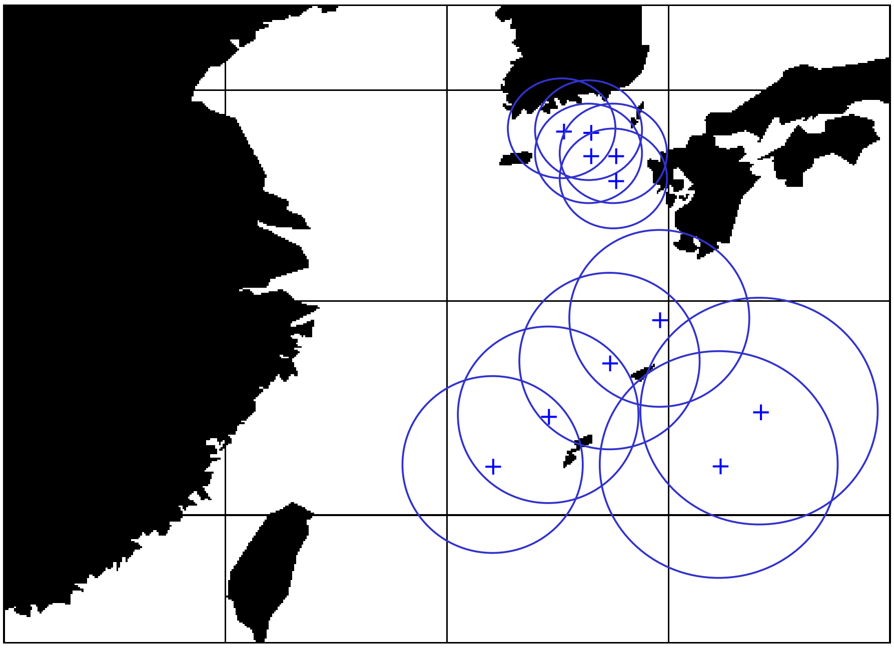

Fig. 2. Simulated scenario theater of operations in the East China Sea showing passive-acoustic sensor locations (+) and approximate areas of sensor coverage (circles)

The theater of operations chosen for the simulation is the East China Sea and vicinity, as shown in Fig. 2. This is an area of potential interest for naval operations and it provides a variety of conditions to test the technologies under development in KMDT. Both shallow-water littoral and deep-water environments are represented, with sea access via straits and open ocean. Mainland and islands about the perimeter of the East China Sea provide for topographical constraints as well as locations for land-based sensor sites, air bases, and ports of opportunity for surface vessels and submarines.

The current stage of the simulation focuses on surface targets, although future simulations may include submarine and air targets as well. Classification of targets can be friendly, hostile, neutral, or unknown. Two motion models are provided: a quasi-random model to represent realistic tracks of unknown targets and a deterministic "waypoint track" model to represent vessels in transit lanes or on patrol. The quasi-random model provides for varying tracks in order to accommodate Monte Carlo analyses. The simulation can represent both fixed- and mobile-sensor platforms. Fixed-sensor platforms will include deep-water arrays, barrier arrays, and land-based systems; mobile sensor platforms include surface vessels and aircraft. The primary detection systems to be modeled are passive acoustic and electromagnetic sensor systems.

The simulation design calls for intelligent agents to access first the static sensor C&P database, which is based on the sensor ontology, to define data requirements, sensor charac-

teristics and performance capabilities. Then agents access the dynamic detection databases, which represent web portals for sensor platforms in the battle space. These web portals are assumed to reside on secure communications network ports. Then information derived from the dynamic detection databases is fused to help reduce uncertainty in the battle space. In a distributed network environment, this fusion could occur either at distributed sites or at a central command center.

4 Static and Dynamic Database Design

The simulation involves both static and dynamic sensor data. Static sensor data describe sensor parameters that remain constant during a mission. Dynamic data include information about the sensors' detections, including the time of detection, location of the sensor, the LOB to the contact, and any signal characteristics such as peak frequencies. Dynamic data also may include sensor state information, such as operating mode or quality of service. The dynamic databases generated by the KMDT simulation are described below.

The integrated sensor ontology and database development for KMDT is described in detail in [3] and [8]. For example, the integrated ontology includes concepts from the standard Naval Tactical Display System Symbology [11], and the VIS sensor ontology [9]. The ontology is the basis for the sensor-model design and the characteristics and performance (C&P) database. The static C&P database consists of data about sensors such as the type of sensor, what the sensor is designed to detect, its detection limits, and constants associated with computing the accuracy of detection. By avoiding transmission of static data over the network, the C&P database mitigates problems associated with limited available bandwidth and repeated exposure of information about sensor capabilities to cryptologic analysis by adversaries.

The detection database is a dynamic repository for all target detections and false alarms generated by the simulation. The simulation design utilizes the database containing information from the respective sensors to represent separate web portals for distributed platforms, as might be the case in operational applications. This design simplifies the connectivity requirements of the network while still representing the functionality inherent in an operational system. This database consists of an ASCII file maintained and updated on the simulation computer. Computers connected to the LAN may access this file. Each record of the database file consists of a single detection alert or a false alarm, with the following parameters:

1. Elapsed time from the start of demonstration run.
2. Platform and sensor identification. Performance parameters corresponding to the sensor are maintained in the C&P database.
3. Sensor location (latitude, longitude). Location may vary if platform is mobile or if the sensor is a remote device (e.g. a satellite) that reports to a shore-based sensor station.
4. Line-of-bearing (LOB) from sensor to contact.
5. Associated source frequencies; multiple frequencies may be observed from one contact.

In addition to the parameters listed above, the following ground-truth information extracted from the simulation is appended to the detection-alert record to validate the performance of the technical approach. Ground-truth, of course, would not be available to an operational user.

6. Target location (latitude, longitude).
7. Target identification (subsurface, surface, air; friendly, hostile, neutral).

The primary detection parameter derived from the simulation is the LOB. To demonstrate a classification capability, the frequencies associated with the LOB detections (available from the scenario definitions) also are reported.

5 Agent Design and Development

5.1 Overview

A generic intelligent agent has a set of goals (or intentions), certain performable actions, and some knowledge (or beliefs) about its environment [6]. The goal of the KMDT agents is to discover and retrieve relevant new information to help identify and correlate sensor contacts. This is accomplished based on calculations the agent performs and decisions it makes to determine the applicable web information to retrieve. The operator can create and deploy several types of agents from the user interface depicted in Fig. 3. The operator specifies LOB search criteria then deploys agents. Search methods enable the operator to override the agent's default inferencing mechanism. As part of the implementation design, all of the KMDT agents have event monitor, event queue, state, timer, and Prolog-like rule processing.

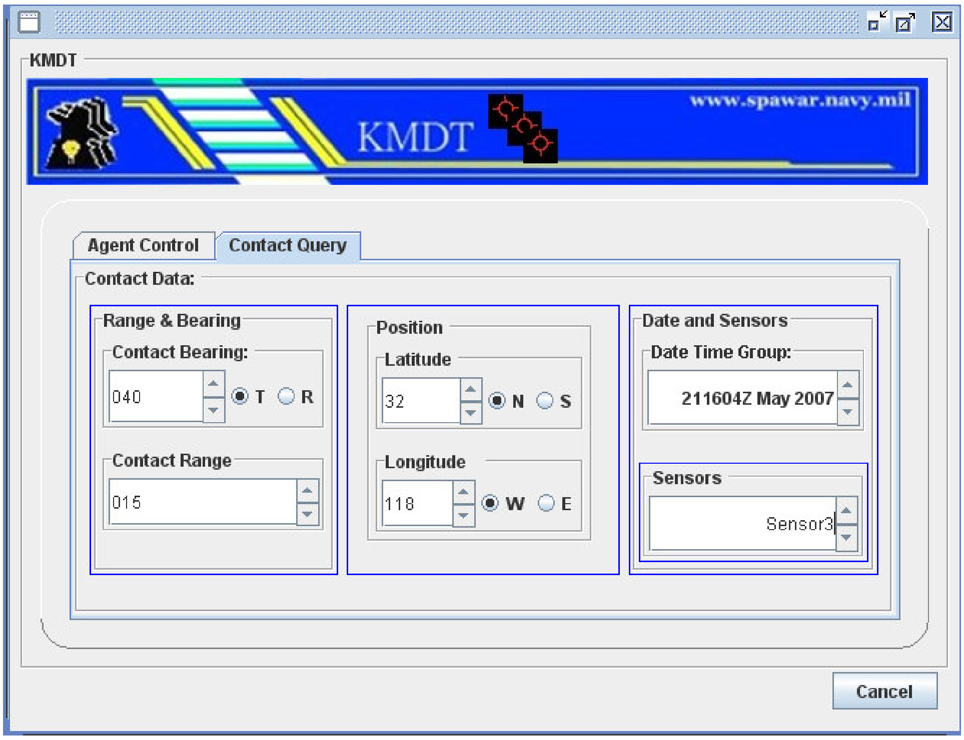

Fig. 3. An operator interface for specifying contact parameters.

Each type of agent is assigned a specific task as depicted in Fig. 1. These agent types are the file agent, which controls and monitors the progress of the other agents, the data-to-database agent, the cross-fix agent, the temporal-proximity agent, and the notification agent. The file-monitor agent sends recently collected sensor data from the M&S scenario to the data-to-database agent. The data-to-database agent stores the data in the contacts database. The cross-fix agent attempts to associate each validated LOB (i.e. not a false alarm) with a crossing LOB from the contacts database. The agents try to classify each contact by comparing source-frequency signature data against that of known targets of interest maintained in appropriate databases. The operator's interface for creating and controlling the agents is shown in Fig. 4, which also shows the activity screens of the file agent and the data-to-database agent as it inserts records into the contact database.

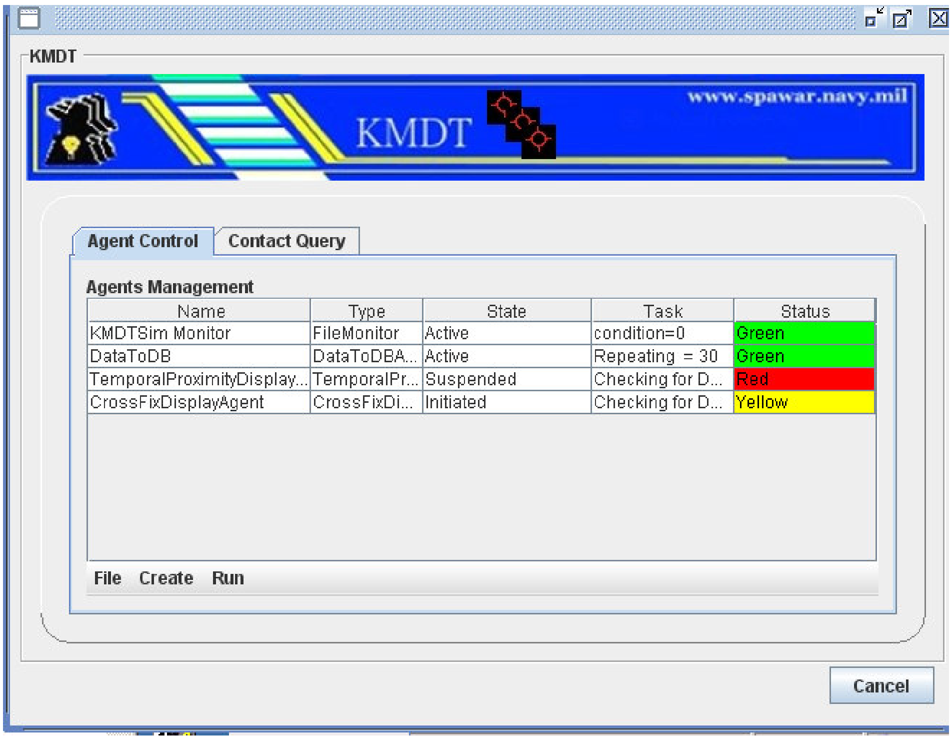

Fig 4. Operator interface for agent control showing data screens to monitor agent activity.

When a request for LOB information is initiated, the agent establishes connectivity with the simulation computer and acquires records from the detection database (a web portal) via the LAN. Based on its knowledge module, the agent decides whether or not the records contain information that might be valuable in satisfying the request.

To accomplish this, especially in the case of heterogeneous fusion, the agent must have access to a sensor ontology and additional databases that contain information on various sensor parameters, classification signatures, data relationships, etc. (See, for example, [7] and [9]). The agent also must have algorithms that govern decisions about whether the acquired information is appropriate. In essence, these algorithms, along with the sensor ontol-

ogy and associated databases, comprise the “intelligence” of the agent. To collaborate with agents on other web portals on the network, agents need explicitly encoded and sharable ontologies.

The design is agent centric in the sense that capabilities are composed of loosely coupled agents that are modified for a specific purpose. The KMDT-agent design was based on designs that were used successfully on past projects. It is constructed from a set of top-level Java classes. These top-level Java classes, along with other support classes, form a framework to build incrementally and add more advanced features based on the research requirements.

5.2 Detailed Design

Agents reside on the command and control computer. When a query is initiated, the agents must perform the following actions:

1. Consult the static C&P database and the sensor ontology to determine data relationships and the appropriate sources of information to query. This action narrows the search to the most likely repositories of information on the LAN. For example, if a sensor system were identified to be incompatible with the data requirements of the task, or if the sensor system were located outside the area of operations, the corresponding web portal is excluded from any search.
2. Establish connectivity over the LAN with web portals that might provide information of interest. The demonstration features LAN connectivity between the command and control computer and the simulation computer to allow access to the detection database.
3. Search the detection database for potential detection alerts that fall within a given time window. A detection alert from one sensor is not likely to match exactly a detection alert from another sensor on the same contact. The time window is determined dynamically by the KMDT agent based upon knowledge about the contact of interest and any inherent systemic latency in sensor reporting. The type of contact, its speed, the detection source type, detection platform motion, and any systemic latency has to be considered to identify potential detection alerts.
4. For potential detection alerts with LOBs that meet the above criteria, the agent determines whether or not the LOBs can intersect within their maximum detection ranges.

When the above criteria have been satisfied, the software agent passes the contact information to the fusion engine, which attempts to associate each detection alert contact with existing tracks on targets of interest. The fusion engine also tries to classify the contact by comparing source-frequency signature data against that of known targets of interest maintained in appropriate databases. For the initial demonstration, hypothetical signatures are constructed to allow for classification of targets as friendly, hostile, or neutral.

6 Software Reuse in Agent Design

KMDT agents are constructed from a set of top-level java classes. These classes were developed and tested on previous projects. Support classes also have been developed and used successfully on past projects. Both the agent and support classes are extensible and comprise the framework and add more functionality incrementally over the life of the KMDT project. Agent communication code rarely is reused project to project. One reason is that the experimental nature of the code itself usually means that it is more efficient to develop code “from scratch” than to customize code from another project. Another reason is that

agent communication in general has become increasingly more complex to implement over time.

KMDT agents communicate over TCP/IP. Before the time of firewalls, e-mail worms, web services, XML, and SOAP, agent communication was more straightforward to implement. Socket connections were made between agents and data were exchanged via Knowledge Query and Manipulation Language (KQML) coordination. This was accomplished using a C language version of Linda. Introduced by Carriero and Gelernter's in 1983, Linda is an elegant, efficient, message-passing coordination technique for distributed processing. The technology has evolved. Communicating via TCP/IP means having to resolve how agents connect to each other, authenticate themselves, encode and decode XML messages, receive messages, and report errors.

The KMDT design incorporates Blocks Extensible Exchange Protocol (BEEP) as a key enabling technology. BEEP is a peer-to-peer protocol framework that solves a number of problems, such as synchronizing messages, managing some parallel operations, and authentication. BEEP has reduced some of the complexity in implementing agent commutations in the KMDT project.

7 Agent-Ontology Interactions

Top-level java classes for constructing KMDT agents were developed with the knowledge that the agents' tasking would involve sensor data processing including but not limited to data retrieval, storage, and fusion. Thus, the agents interact with an integrated sensor ontology [7], [8], [9] to coordinate their tasks either directly or indirectly through databases designed according to the ontology. (See section 5.) The ontology and associated databases document the sensor and platform characteristics and capabilities. They inform the agents of the appropriate sources of data for a particular task so that the agent will search only for information relevant to the task. Agents might consult the ontology to identify classes of sensors that can detect a certain type of contact, thus limiting the subsequent search space to include only those sensor types. It would be unproductive, for example, to search web sites of space-based sensors for information on an underwater acoustic contact.

The sensor ontology and associated databases also supply knowledge to the agents to aid subsequent data fusion. If two sensors of the same type (homogeneous sensors) detect the same signal, their signal signatures can often be compared directly. However, data fusion for dissimilar (heterogeneous) sensors is more difficult and less direct because a direct comparison of the signal signatures is usually not possible. In this case, an ontology is especially useful. If an unknown contact is detected by disparate sensors, agents can use the ontology to identify classes of TOI that can be detected by each sensor type, whose intersection is likely to contain the unknown contact, thus aiding in subsequent classification. As a simple example, suppose an underwater acoustic sensor detected some unknown contact at the same time a space-based electromagnetic sensor also detected an unknown contact. Knowledge extracted from the ontology that the likely acoustic contact was either an undersea or surface target, whereas the electromagnetic contact was either a surface or air target, would imply that the contact was a surface target if the respective detections were correlated.

8 Results

The file agent and database-interface agent transfer and store the detection data. Then, the cross–fix agent accesses the detection information directly from the detection database. The cross–fix agent transforms each detection from a location point into a vector, then examines other detections for possible a cross-fix. A typical result of the cross-fix agent is depicted in Fig. 5. This display originally was developed as a debugging tool.

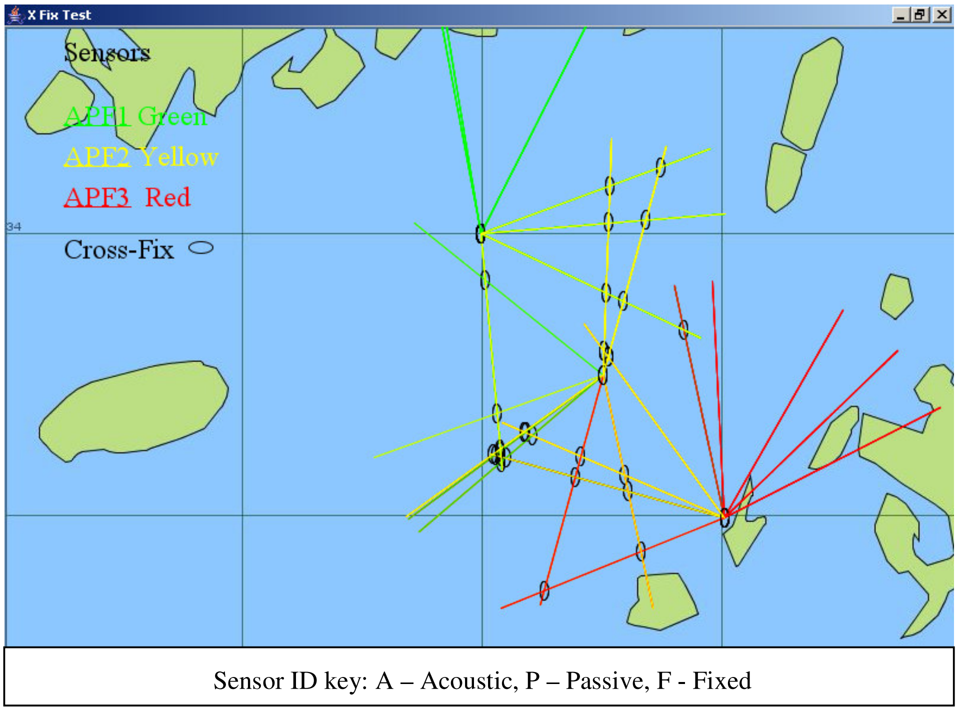

Fig. 5. Results of agent computations showing the LOB cross fixes as ovals.
(For sensor locations, see Fig. 2.)

9 Directions for Future Research, Development, and Applications

Experience and preliminary research results suggest the evolving KMDT technology can provide the military users the following improved capabilities:

- Agents and ontologies to identify, acquire, and analyze track information.
- Tools for transforming and disseminating relevant track information to the commander in sufficient time to act.
- A flexible and operationally relevant human-computer interface supporting analytical and intuitive styles of queries.
- Reduced uncertainty in command decisions based on a more efficient use of data from existing sensors that are better correlated and utilized in a net-centric environment.

- More accurate level-one sensor fusion.
- Ability to deploy agents using "point and click" while wearing protective clothing.

In the present design, agents acquire message-level data from remote web portals to be fused at the site that deployed the agents. In contrast, a future design could require the agents to process the message-level data from the web portals remotely at the site from which the data are retrieved. This distributed architecture design reduces network bandwidth requirements. This processing method is designed to relieve overloaded operators tracking multiple unknown contacts and who may have deployed several agents to retrieve data on each one [3]. Future simulations will include heterogeneous sensor models. Future simulations also could include active systems in addition to passive acoustic or electromagnetic sensors.

The KMDT architecture fits into the larger overall design of net-centric web services for the war fighter. KMDT LOB correlations can be orchestrated to support individual users and their software agents in the net-centric environment where the next service utilization depends on the outcome of current services. Good orchestration requires good semantic understanding of services through ontologies [12].

Acknowledgements. The authors thank Ms. Brenda J. Powers for enlightening discussions regarding agent technology. The authors thank the Office of Naval Research and the Space and Naval Warfare Systems Center, San Diego, Science and Technology Initiative for their support of this work. This paper is the work of U.S. Government employees performed in the course of employment and no copyright subsists therein. It is approved for public release with an unlimited distribution.

References

1. Clark, V. ADM, USN: Sea Power 21 Series – Part I: Projecting Decisive Joint Capabilities. Naval Institute Proceedings **128** Oct. (2002) 32-41
2. McGirr, S.C.: Resources for the Design of Data Fusion Systems," Proceedings of the ISIF Fusion 2001 Conference on Information Fusion, Montreal, Canada, Aug. (2001)
3. Ceruti, M.G., Wright T.L., Wilcox, D.R., McGirr, S.C.: Modeling and Simulation for the Knowledge Management for Distributed Tracking (KMDT) Program. Proceedings of the International Workshop on Modeling and Applied Simulation (MAS-2005). Oct. (2005) 67-75
4. Ceruti, M.G.: Mobile Agents in Network-Centric Warfare. Institute of Electronics Information and Communication Engineers Transactions on Communications. Tokyo, Japan. **E84-B**. No. 10 Oct. (2001) 2781-2785
5. Ceruti, M.G., Powers, B.J.: Intelligent Agents for FORCEnet: Greater Dependability in Network-Centric Warfare. Supplemental Volume of the Proceedings of the IEEE International Conference on Dependable Systems and Networks (DSN 2004). June (2004) 67-75
6. Deazeau Y., Muller, J-P.: Decentralized Artificial Intelligence. Proceedings of the 1st European Workshop on Modeling Autonomous Agents in a Multi-Agents World. Elsevier Science, Cambridge, England: (1989) 3-13
7. Ceruti, M.G.: Ontology for Level-One Sensor Fusion and Knowledge Discovery. Proceedings of the 2004 International Knowledge Discovery and Ontology Workshop (KDO-2004). Sep. (2004) 20-24
8. Ceruti, M.G., Wilcox, D.R.: Sensor Ontology Integration for the Knowledge Management for Distributed Tracking (KMDT) Program. Proceedings of the 11th Command and Control Research and Technology Symposium (CCRTS 06). June (2006)
9. Matheus, C.J., Tribble, D., Kokar, M.M., Ceruti, M.G., McGirr, S.C.: Towards a Formal Pedigree Ontology for Level-One Sensor Fusion. Proceedings of the 10th International Command and Control Research and Technology Symposium. June (2005)

10. Waltz, E., Llinas, J.: Multisensor Data Fusion. Artech House, Boston (1990)
11. Lead Standardization Activity (LSA). Center for Standars (CFS), Department of Defense Interface Standard Common Warfighting Symbology. MILSTD2525B. Reston, VA, USA. Jan. (1999)
12. Daconta, M.C., Obrst L.J., Smith, K.T.: The semantic Web: A guide to the future of XML, Web Services, and Knowledge Management. Wiley Publications, Indianapolis, Indiana (2003) 72-75